#6

(12) UK Patent Application (19) GB (11) 2 352 064 (13) A

(43) Date of A Publication 17.01.2001

(21) Application No 9916289.3

(22) Date of Filing 13.07.1999

(71) Applicant(s)
Thomson Training & Simulation Limited
(Incorporated in the United Kingdom)
Gatwick Road, CRAWLEY, West Sussex, RH10 2RL, United Kingdom

(72) Inventor(s)
David John White
Alan Edward Ball
John David Haughton

(74) Agent and/or Address for Service
Marks & Clerk
Sussex House, 83-85 Mosley Street, MANCHESTER, M2 3LG, United Kingdom

(51) INT CL^{7}
G06F 13/40 15/16

(52) UK CL (Edition S)
G4A AFGDC

(56) Documents Cited
GB 2329984 A **WO 97/19405 A** **US 5848249 A**

(58) Field of Search
UK CL (Edition R) **G4A AFGDC AFGL AMP**
INT CL^{7} **G06F 13/38 13/40 15/16**
Online: WPI, EPODOC, PAJ, INSPEC, COMPUTER

(54) Abstract Title
Multi-processor system with PCI backplane

(57) A multi-processor computer system comprises one or more computer modules (1) connected to a host computer via a common PCI bus system backplane. Each computer module (1) comprises a PCI connector connected to the PCI backplane at least one CPU, a PCI backplane interface (10) installed between the PCI connector and the or each CPU (5) for forwarding PCI memory cycles to and from the PCI backplane, memory address translating means (11) for translating memory cycles forwarded from the PCI backplane. Each CPU comprises a local microprocessor (5) having an associated local memory unit (8), and a bridge (7) The CPU bridge (7) includes an address decoder which is programmed to a local address range for access to the respective computer module address space including the local memory unit (8).

The PCI interface is programmed to forward PCI memory cycles from the PCI backplane which fall within an address range which is different from the local address range of the or each CPU (5) of that module and which is different for each computer module (1) in the system. The translating means (11) operates to translate memory cycles falling within its address window into corresponding addresses of the local address space of the respective computer module (1) such that addresses corresponding to a local memory unit of that module may be decoded by the respective CPU bridge decoder. The result is each microprocessor of the system can access the entire PCI memory space comprising the memory space of each computer module (1) including the local memory units of all microprocessors of the system.

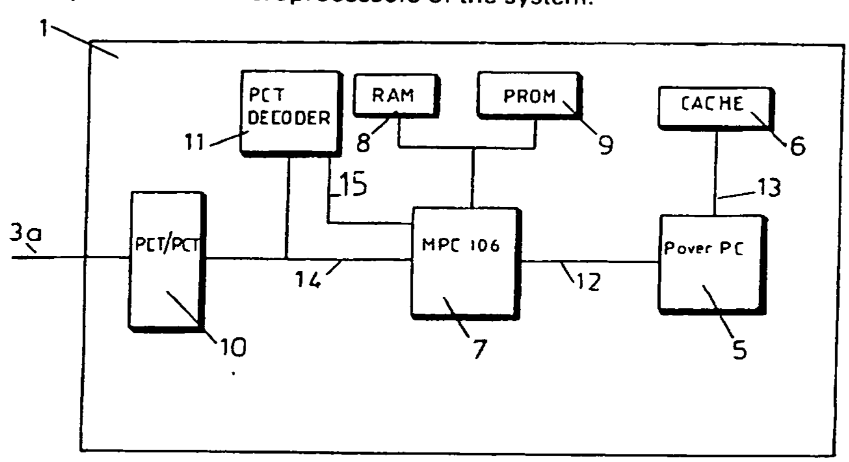

FIG. 2

At least one drawing originally filed was informal and the print reproduced here is taken from a later filed formal copy.

GB 2 352 064 A

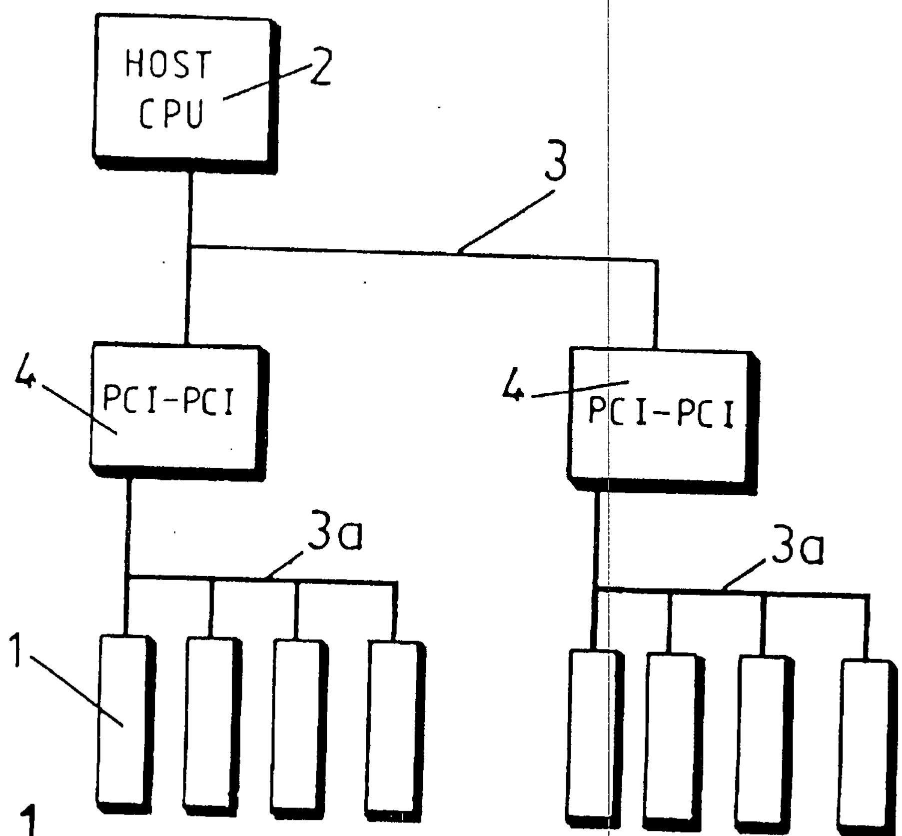

FIG. 1

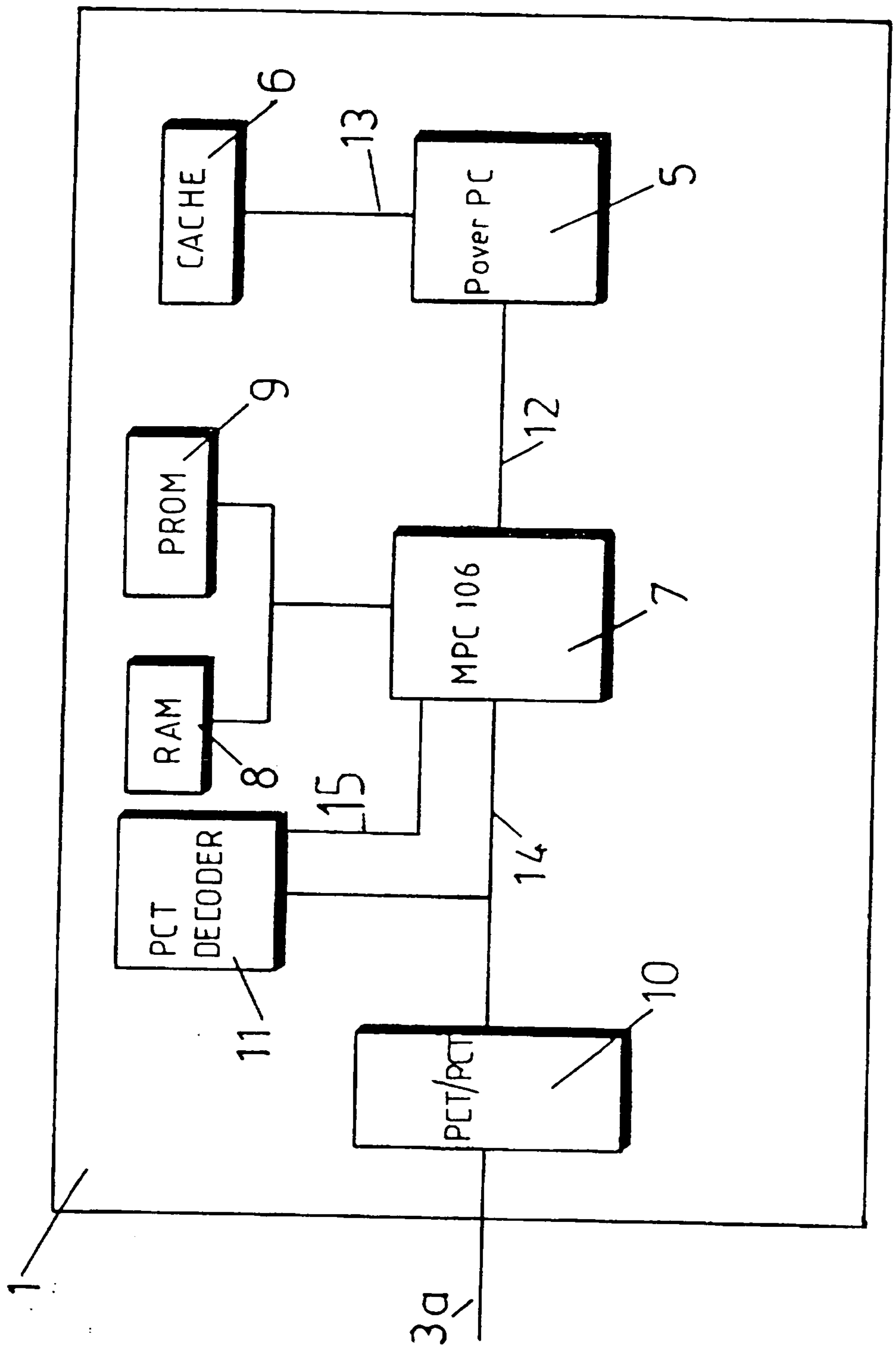
CACHE
6
13
Pover PC
5
PROM
9
12
MPC 106
7
RAM
8
15
PCT
DECODER
14
11
10
PCT/PCT
1
3a

FIG. 2

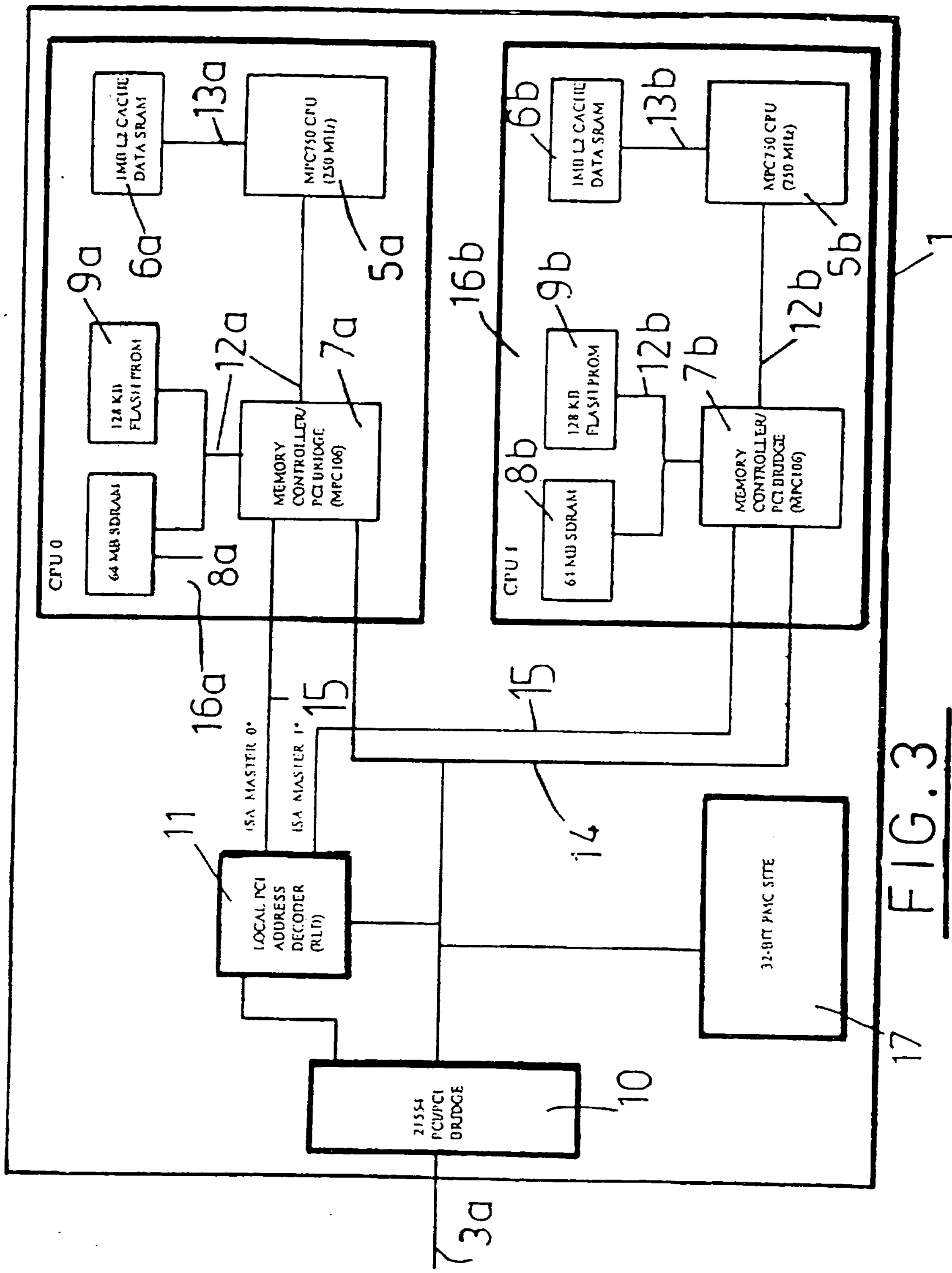
CPU 0
1MB L2 CACHE DATA SRAM
MPC750 CPU (250 MHz)
128 KB FLASH PROM
64 MB SDRAM
MEMORY CONTROLLER/ PCI BRIDGE (MPC106)
CPU 1
ISA MASTER 0
ISA MASTER 1
LOCAL PCI ADDRESS DECODER
32-BIT PMC SITE
21554 PCI/PCI BRIDGE
1
3a
5a
5b
6a
6b
7a
7b
8a
8b
9a
9b
10
11
12a
12b
13a
13b
14
15
16a
16b
17

FIG. 3

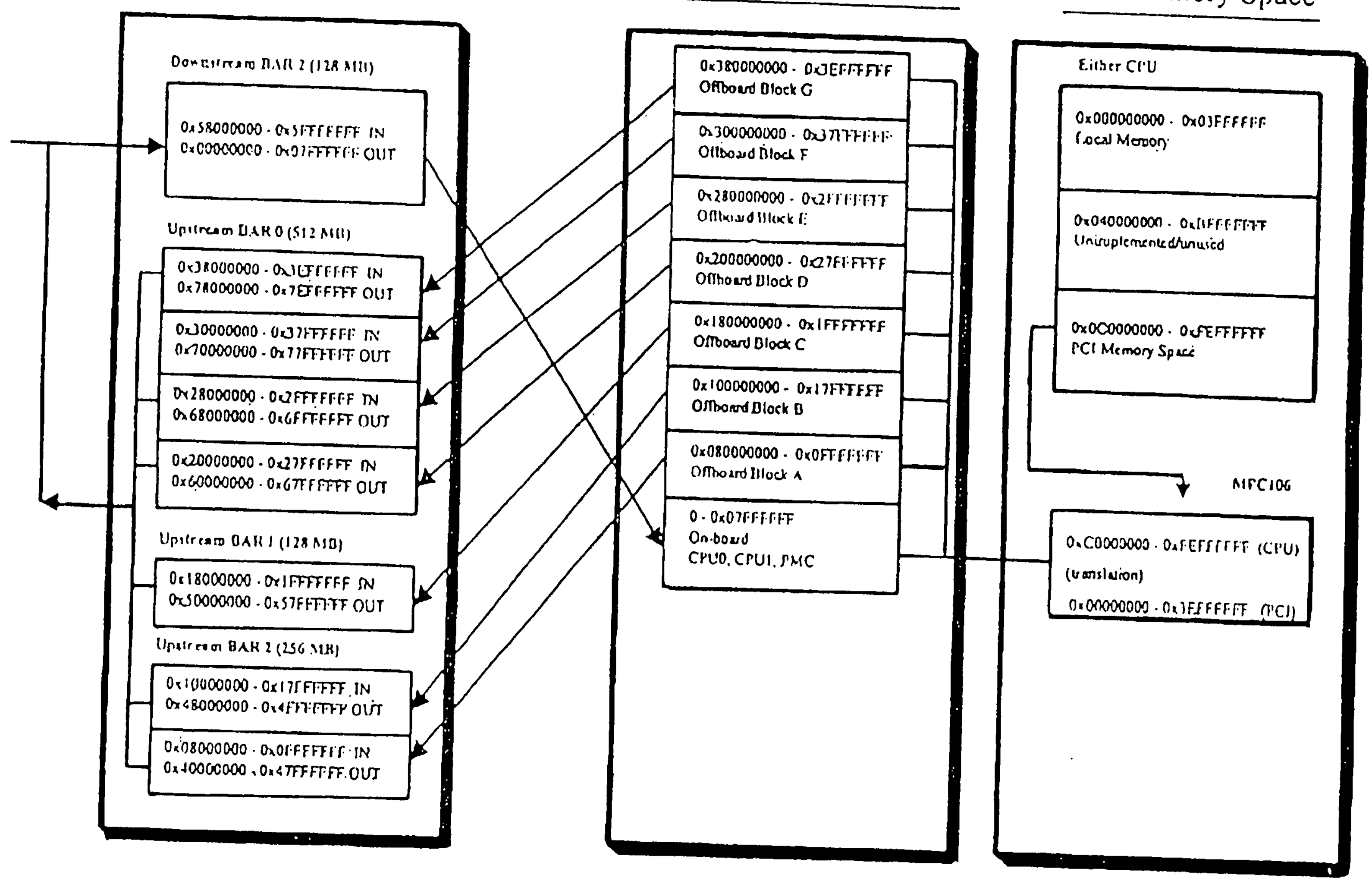
Host PCI
21554 Bridge
Local PCI Memory Space
CPU Memory Space
Downstream BAR 2 (128 MB)
0x58000000 - 0x5FFFFFFF IN
0x00000000 - 0x07FFFFFF OUT
Upstream BAR 0 (512 MB)
0x38000000 - 0x3FFFFFFF IN
0x78000000 - 0x7FFFFFFF OUT
0x30000000 - 0x37FFFFFF IN
0x70000000 - 0x77FFFFFF OUT
0x28000000 - 0x2FFFFFFF IN
0x68000000 - 0x6FFFFFFF OUT
0x20000000 - 0x27FFFFFF IN
0x60000000 - 0x67FFFFFF OUT
Upstream BAR 1 (128 MB)
0x18000000 - 0x1FFFFFFF IN
0x50000000 - 0x57FFFFFF OUT
Upstream BAR 2 (256 MB)
0x10000000 - 0x17FFFFFF IN
0x48000000 - 0x4FFFFFFF OUT
0x08000000 - 0x0FFFFFFF IN
0x40000000 - 0x47FFFFFF OUT
0x380000000 - 0x3EFFFFFF
Offboard Block G
0x300000000 - 0x37FFFFFF
Offboard Block F
0x280000000 - 0x2FFFFFFF
Offboard Block E
0x200000000 - 0x27FFFFFF
Offboard Block D
0x180000000 - 0x1FFFFFFF
Offboard Block C
0x100000000 - 0x17FFFFFF
Offboard Block B
0x080000000 - 0x0FFFFFFF
Offboard Block A
0 - 0x07FFFFFF
On-board
CPU0, CPU1, PMC
Either CPU
0x000000000 - 0x03FFFFFF
Local Memory
0x040000000 - 0x0FFFFFFF
Unimplemented/unused
0x0C0000000 - 0xFEFFFFFF
PCI Memory Space
MPC106
0xC0000000 - 0xFEFFFFFF (CPU)
(translation)
0x00000000 - 0x3EFFFFFF (PCI)

FIG. 4

A MULTI-PROCESSOR COMPUTER SYSTEM

The present invention relates to a multi-processor computer system.

Multi-processor computer systems which provide increased processing power through parallel processing operation are known. Such systems are used in a wide variety of applications such as, for example, flight simulators wherein simulator functions are allocated to different processors. For instance, one known flight simulator system comprises a number of commercial "off-the-shelf" single board computer (SBC) real-time TARGETS each based on a Motorola PowerPC microprocessor, connected to a Data Processing HOST comprising of another Motorola PowerPC SBC. In this system, the HOST and TARGETS are interconnected by a VME 32-bit bus system which is one of a number of known multi-processor bus systems. PowerPC CPUs are used because commercial avionics simulation code is big-endian and thus cannot be run on an Intel x86 processor (or clone).

In more detail, each single board computer comprises a PowerPC microprocessor with associated non-volatile programme memory , system memory (DRAM), and level 2 cache memory, interconnected by a standard PowerPC local bus. A bridge is provided from the PowerPC local bus to an on-board peripheral expansion bus, complying with the Peripheral Component Interconnect (PCI) bus standard. Control of memory systems on the PowerPC local bus as well as the bridge between PowerPC local bus and PCI bus is provided by a Motorola MPC105/6 PCI Bridge/Memory Controller. Connected to the PCI bus are ethernet and graphics adapters and an SCSI-2 interface (an industry standard Small Computer System Interface providing for the connection of a variety of peripheral devices). An ISA (Industry Standard Architecture) bus is connected to the PCI bus via a PCI-ISA bridge and provides facilities for the interconnection of I/O devices such as mouse, keyboard, floppy drives and serial ports. A further bridge provides connection from the PCI bus to the off-board (back-plane) VME bus for communication between SBC's.

Full technical specifications are readily available for all the above mentioned components and bus architectures.

Whilst this known flight simulator system has the benefit that it is constructed from readily available commercial-off-the-shelf components, it does have a number of disadvantages. Limitations to the clock rate and bandwidth of the VME bus are such that inter-processor communication via the VME bus is significantly slower than the intra-processor performance and thus the VME bus presents a communications bottleneck. In addition, VME based systems present scalability problems since the bus may be saturated by a relatively small number of processors. One means of increasing the total number of available interconnected processors is to employ multiple VME sub-systems, together with some means of communicating between sub-systems, typically some form of reflective memory. Such a system can be designed in such a way that the majority of inter-processor communication remains within a sub-system, and has no impact on other sub-systems, whilst the reflective memory system provides communication between sub-systems when required. This approach however is relatively expensive in terms of the additional VME packaging hardware, support hardware and the reflective memory system required.

It is an object of the present invention to provide a multi-processor computer system which obviates or mitigates the above disadvantages.

The PCI system architecture provides advantages over the VME bus based prior art system, particularly in that it has a greater bandwidth and this provides higher inter-processor communication speed. For instance, a 32 bit PCI bus operating at 33 MHz can transfer data at a rate of 132 Mbytes/Sec. Furthermore, 64 bit extensions (with operation speeds of 66 MHz) are now available.

The present invention may be implemented using only commercially available components in order to minimise production costs, and can be implemented using conventional components in a non-conventional manner. In particular, the invention makes it possible to provide a system architecture and configuration software which enables PowerPC processors to be linked to a host computer via a common PCI bus back-plane in such a way that all processors in the system have access to the entire PCI memory space (i.e. individual PowerPC processors have access to both their own local memory and to the local memories associated with all the other processors in the system).

Specific embodiments of the present invention will now be described, by way of example only, with reference to the accompanying drawings, in which:

Fig.1 is a schematic illustration of a multi-processor computer system in accordance with the present invention;

Fig. 2 is a schematic illustration of the architecture of a single board CPU in accordance with a first embodiment of the present invention;

Fig. 3 is a schematic illustration of the architecture of a single board CPU in accordance with a second embodiment of the present invention; and

Fig. 4 illustrates an example of a memory addressing/translating scheme suitable for the second embodiment of the invention illustrated in Fig.3.

Referring to Fig. 1, this schematically illustrates the basic architecture of one possible computer system according to the present invention based on PCI system architecture and comprising eight single board PowerPC microprocessor based single board computers (SBCs) 1 connected to an x86 microprocessor-based host computer 2 on a common PCI bus back-plane 3/3a (where 3 and 3a are primary and secondary PCI buses respectively). In the illustrated example, the host computer 2 is a conventional PC which has at least eight PCI expansion slots. PCI expansion slots are arranged in groups of four, each group connected to the host, and to the other group, via PCI-PCI bridges, which are an integral part of the back-plane 3/3a. Each group of four SBCs 1 can thus communicate amongst themselves, or with the host computer 2, without impacting the other group.

It will be readily appreciated that since the topology of the PCI system is a bridge architecture, additional PowerPC SBCs 1 could be installed if further PCI expansion slots were available, without significantly impacting communication between SBCs within a group. It will also be appreciated that fewer than eight PowerPC SBCs may be installed, and indeed only a single PowerPC SBC may be installed. Versatility and scalability provided by the PCI bridge based architecture is one of a number of advantages of the present invention over existing VME based systems.

Referring now to Fig. 2, this illustrates the major components and basic architecture of a relatively simple SBC 1 in accordance with a first embodiment of the present invention. The illustrated SBC 1 comprises a PowerPC processor 5 with

dedicated Level 2 cache memory 6, a Motorola MPC106 bridge/memory controller 7, a main memory unit 8, a boot ROM unit 9, an Intel 21554 PCI-PCI bridge 10 and a dedicated local PCI address decoder 11.

The processor 5 is installed on a local PowerPC main memory bus 12 and is connected to the cache memory unit 6 by a cache bus 13. The MPC106 bridge 7 provides both control of the local memory unit 8 and a bridge between the main memory bus 12 and a local on-board PCI bus 14. The decoder 11 is installed on the local PCI bus 14 and the PCI-PCI bridge 10 provides a bridge between the local PCI bus 14 and the host PCI back-plane 3/3a.

It will be understood that the SBC 1 will also include other conventional components such as an on-board power supply and the necessary PCI connectors to enable the board to be connected into the host PCI back-plane 3/3a. Such components, which are not essential to an understanding of the implementation of the present invention, will not be described in detail here. The SBC 1 may, for instance, be configured as a full length (312mm) PCI mezzanine card.

The Motorola MPC106 bridge/memory controller 7 includes a PowerPC processing interface, a memory interface, a PCI interface and four address registers (only one of which is used in the present invention). The PCI interface (which connects the local main memory bus 12 to the local PCI bus 14) is compliant with the PCI local bus specification, revision 2.1, and supports access to all PCI address spaces. The MPC106 7 functions both as a PCI bridge and as a memory controller. As a PCI bridge, the MPC106 allows the local processor 5 to access other PCI agents (in such cases the MPC106 is functioning as a PCI Master). As a memory controller, the MPC106 provides the necessary timing and control to allow either a local processor or another PCI agent (via the MPC106 bridge function) to access the system memory 8.

The Intel 21554 PCI-PCI bridge 10 is a non-transparent bridge specially designed for intelligent embedded boards. Since the bridge 10 is non-transparent, the primary bus (which in this case is the PCI back-plane 3/3a) cannot by default detect the secondary bus (which in this case is the local PCI bus 14) and is not therefore required to configure any PCI devices on the secondary bus. The PCI-PCI bridge 10 has true base address registers for forwarding PCI memory accesses in both directions

and is capable of address translation so that an address appearing on the primary interface from the PCI back-plane 3/3a can be translated before it is issued on the local PCI bus 14.

The decoder 11 is implemented using a conventional PCI-compliant programmable logic device, programmed to decode PCI addresses at the start of each PCI transaction on the local PCI bus 14.

It will be seen immediately that the single CPU board comprises readily available commercial off the shelf components (full technical specifications of which are readily available) and moreover comprises only a minimum number of components thereby reducing cost. For instance, the board does not comprise any PCI adapters (such as graphics, ethernet and SCSI adapters) or an ISA bus and associated adapters which are not required to be provided by the CPUs 1 because they will be provided by the host computer 2. This is a further advantage of the present invention over prior art systems mentioned above.

Other important features of the various devices will become apparent from the following description of the operation of the SBC 1.

The system architecture and configuration software are designed so that each processor 5 has access not only to its dedicated memory unit 8 but also to the entire PCI memory space including the memory of every SBC 1 (via the PCI bus). Achieving this using only commercially available components not specifically designed for this purpose is a problem addressed by the present invention. In particular the MPC106 bridge 7 does not directly allow local CPU memory 8 to be mapped into PCI memory space (essentially because as a host bridge it regards local memory and PCI memory as separate spaces). The invention solves this problem by using each MPC106 bridge chip in conjunction with the PCI-PCI bridge 10 and PCI decoder 11, in a non-conventional way.

The processor 5 of each SBC 1 can generate memory accesses which must be translated either as local memory accesses or accesses to the PCI memory space which must include the local processor memory units 8 and local PCI spaces of each other SBC 1 in the system. Thus, each physical memory unit 8 of each SBC 1 must be accessible from its local processor 5 via a first address range, and also accessible by any other off-board processor 5 via a second address range.

The MPC106 bridge 7 of each SBC 1 is programmed to operate conventionally in decoding local memory and PCI access from the respective local processor 5. The appropriate MPC106 decoder may be programmed to the same predetermined start address on each CPU board 1 so that software code (which is loaded from the host computer during configuration and initialisation) does not have to be specific to any particular board (i.e. would not need to be relocated depending on which board it is run) but rather can be run on any CPU 1. However, since the MPC106 decoder is programmed to an appropriate address range for local memory access from the on-board processor 5 it cannot be used directly for access to respective local memory units 8 from an off board CPU (i.e. another SBC 1) via the PCI back-plane. In addition, the MPC106 PCI memory space window is hard-wired and is not programmable.

Accordingly, a single downstream Base Address Register (BAR) of the PCI-PCI bridge 10 is programmed by a host configuration software to provide access to the memory space of the associated SBC 1, comprising the local PCI space and local main memory space. The memory window programmed into the BAR of the PCI-PCI bridge 10 by the host configuration software will be different for each SBC 1 depending on the order in which each SBC is identified by the host. The PCI-PCI bridge is programmed to translate memory accesses generated off-board and falling within the respective memory address window into appropriate local addresses for accessing the physical memory available on that board. There is, however, a further problem in that, as mentioned above, the MPC106 bridge 7 does not directly allow PCI memory cycles generated off-board to be decoded as local memory space cycles. Thus, translation of the address range by the PCI-PCI bridge is not alone sufficient to allow access to the local memory 8 via the MPC106 bridge 7.

This problem is solved by the provision of the decoder 11 which on identifying a PCI address corresponding to the local memory unit address range asserts signal ISA MASTER (which is side-band signal 15 in Figs 2 and 3) on the MPC106 bridge 7 which causes the bridge 7 to respond to all incoming PCI cycles. That is, with ISA MASTER (i.e. tied low) all PCI cycles appearing at the PCI interface of the MPC106 bridge 7 are decoded as local memory accesses.

Thus, the combination of the PCI-PCI bridge 10, the decoder 11, and the MPC106 bridge 7 operating in accordance with the present invention allows the same physical memory local to a particular SBC 1 to be accessed from both the on-board processor 5 (generating memory accesses within a first address range) and also from an any off-board processor 5 (generating memory accesses within a second address range corresponding to the address window of the PCI-PCI bridge 10 of the target SBC 1).

The SBC illustrated in Fig. 1 is only a relatively simple embodiment of the present invention. It is a particularly advantageous feature of the present invention that the basic architecture allows additional PCI devices, and additional processors, to be located on a single local PCI bus 14. For instance, the architecture of an SBC 1 in accordance with a preferred embodiment of the present invention is illustrated in Fig. 3. As with the SBC 1 of Fig. 2, only components necessary for an understanding of the implementation of the invention are described in detail. In addition, like reference numerals are used for like components.

Referring to Fig. 3, the illustrated SBC 1 comprises two identical CPU blocks 16a and 16b. Each CPU block 16a/16b comprises a PowerPC processor 5a/5b, a cache memory unit 6a/6b, an MPC106 bridge 7a/7b, a main memory unit 8a/8b and boot ROM 9a/9b. Both CPU blocks 16a and 16b are installed on a common local PCI bus 14 which in addition to the PCI-PCI bridge 10 and decoder 11 supports a 32-Bit PCI mezzanine card site 17.

With the SBC of Fig. 3, each processor 5a/5b must have access to its own local memory unit 8a/8b as well as to the local memory unit 8a/8b of the other on-board processor 5a/5b and to the entire PCI space including the memory space of other SBCs in the system. Assuming there is a total of eight SBCs 1 in the system, an example of an address scheme which enables the required memory accesses in accordance with the present invention is shown below in table 1 which shows the mapping of local (i.e. on-board) PCI space of one SBC 1 of the system.

Local PCI Address Range	Target	Size	Notes
0x00000000 – 0x03FFFFFF	CPU 0 Memory	64MB	
0x04000000 – 0x06FFFFFF	CPU 1 Memory	48 MB (e.g.)	(i)
0x07000000 – 0x07FFFFFF	PMC	16 MB (e.g.)	(i)
0x08000000 – 0x0FFFFFFF	(Offboard A)	128MB	
0x10000000 – 0x17FFFFFF	(Offboard B)	128MB	
0x18000000 – 0x1FFFFFFF	(Offboard C)	128MB	
0x20000000 – 0x27FFFFFF	(Offboard D)	128MB	
0x28000000 – 0x2FFFFFFF	(Offboard E)	128MB	
0x30000000 – 0x37FFFFFF	(Offboard F)	128MB	
0x38000000 – 0x3EFFFFFF	(Offboard G)	112MB	
0x3F000000 – 0x3FFFFFFF	Not mapped	16MB	(ii)

TABLE 1

Notes to table:

(i) The split of the upper 64MB between CPU1 and PMC will be configurable – see below.

(ii) The uppermost 16MB of CPU address space is interpreted by the MPC106 as boot ROM access, so accesses do not generate PCI memory space cycles and the space is effectively not useable by any CPU.

Each processor 5a/5b accesses the above local and PCI space via addresses (in the range 0xC0000000 – OxC8xxxxxx) translated down to the above space by the appropriate decoder of the associated MPC106 bridge operating conventionally as mentioned above. The mapping of the off-board space assumes the presence of seven other similar SBCs 1 (address blocks A through G), with the proviso that Block G has 16 MB less accessible space than all the others because of the boot ROM issue noted above.

The PCI-PCI bridge 10 of each SBC 1 is programmed by configuration software to decode addresses within an address window corresponding to the size of the memory block of the respective SBC 1 (which in each case with the described

example is 128 MB with the exception of Board G which has 112 MB). The respective PCI-PCI bridge translates addresses within the associated address window to addresses corresponding to PCI and memory addresses local to that board. Thus, the translated address may be an address corresponding to the memory unit 8a, the memory unit 8b, or the PCI card site 17. The dedicated PCI decoder 11 decodes the translated PCI cycles and if the translated address corresponds to the memory unit 8a asserts ISA MASTER on MPC106 bridge 7a. Alternatively, if the translated address corresponds to the memory unit 8b then the decoder 11 asserts ISA MASTER on MPC106 bridge 7b. Thus, the decoder 11 effectively implements separate PCI address windows for the two MPC106 bridges 7a and 7b as required. If the translated address falls within the range allocated to the PCI card site 17, then neither ISA MASTER signal is asserted and the cycle is claimed by whatever card is installed in the site 17.

For a PCI memory cycle originating on-board, from, say, CPU block 16a, the MPC106 bridge 7a translates the memory cycle and issues it onto the local PCI bus 14. If the translated address corresponds to the PMC site 17 then the cycle is claimed by the PMC device. If, however the translated address corresponds with an address within the address range of the memory unit 8b of the other on-board CPU block 16b, the decoder 11 asserts ISA MASTER on MPC106 bridge 7b. Alternatively, if the translated address corresponds with an off-board address, no ISA MASTER signal is asserted and the cycle may be claimed by the PCI-PCI bridge and issued onto the PCI back-plane 3/3a. It will be appreciated that memory cycles generated from CPU 5b are essential handled in the same way so that CPU 5b has access to memory 8a, PMC site 17 and of board addresses.

It should be noted that since the decoder 11 only monitors PCI signals (apart from the ISA MASTER signals which are side band signals) it will not violate the PCI specification in any way. The ISA MASTER signals do however need to be asserted with the same timing as other PCI signals.

The address translation scheme described above is illustrated in Fig. 4.

Referring to Fig.4, the base address for the entire SBC 1 within host PCI memory space will be determined by programming configuration registers in the PCI-PCI bridge. In particular, a single downstream BAR is programmed to provide access

to the 128 MB block occupied by that particular SBC. The address window must be located on a (module window-size) address boundary. The particular downstream BAR used should be BAR 2 or 3 (in the illustrated example that is BAR 2).

The digital 21554 PCI-PCI bridge 10 provides a total of three BARs for upstream memory space decoding, all of which are used in the illustrated example. Note that because of the requirement that address windows be sized in binary multiples, and aligned on the same size boundary, three BARs are in fact required to fully map seven SBC memory blocks or 128 MB.

The programming shown in Fig. 4 will clearly be different for each SBC, according to what downstream address it occupies.

Access to CSRs is required from the primary side of the PCI-PCI bridge 10 to allow the host to configure the amount of memory space occupied by the PCI card site 17 (using the parallel ROM interface). CSRs can only be accessed in primary PCI memory space using downstream BAR 0, which is why BAR 0 is not used for the translation mentioned above. Alternatively, the host may use PCI I/O space to access CSRs.

The CPU blocks (which in the illustrated example number two) which share a single SBC are specially related in address terms (in order to comply with the illustrated address translation map). Each CPU block will thus need to know whether it is addressing a CPU block on the same board or an off-board CPU block in order to form the correct base address. Each CPU block in the system must therefore have a unique base address table. It will be appreciated that by adopting a different address translation scheme, SBCs in accordance with the present invention may comprise more than two processor blocks and/or PCI devices.

In addition, it should be noted that the translation feature of the Intel 21554 PCI-PCI bridge can be used to locate the 1 GB space occupied by all boards on any 1 GB boundary in host PCI memory space, so no space need be wasted due to host system memory. Furthermore, it is thus possible to install two or more independent x86 multi-processor environments, each comprising eight SBCs, in the same host system, by locating them on different 1 GB boundaries in host PCI memory space. In this sense, the term "environment" refers to the host software driving the multi-processor system. It may be noted that for a 32 bit PCI bus there is only 4 GB of

addressable PCI space and thus, bearing in mind that at least part of the first Gigabyte is required by the host, the practical limit for the number of environments within a single host system is three.

It will be appreciated that details of components such as the particular PowerPC processor 5, and memory units etc may vary from one embodiment of the invention to the next. Examples of some preferences are given below.

The preferred PowerPC processor is the Motorola MPC750 processor.

The main memory bus 10 clock frequency is preferably set to 83.3 MHz, which is the fastest SYSCLK frequency currently supported by both the MPC750 processor 8 and the MPC106 bridge chip 9. Use of the fastest possible SYSCLK maximises main memory performance. The processor clock speed is derived from the memory bus frequency and must use one of the available multiplier ratios. Preferably the ratio, which is selected via external strapping, should be 3.5 which provides a processor clock speed of 291.5 MHz. Similarly, the frequency of the level 2 cache bus 14 (L2CLK) is derived by dividing down the processor clock speed by one of the available divisors (1, 1.5, 2, 2.5 and 3). Due to speed limitations of currently available components the divisor uses 2.5 so that the frequency of the cache bus 14 is set to 116.6 MHz. The SYSCLK to CPUCLK multiplier is set by pull-up and pull-down resistors which can be moved to select different ratios. It will be appreciated that the above clock speeds are preferred speeds based upon the availability of 300 MHz parts.

The preferred PCI bridge (for operation with the above clock frequency scheme) is the 83.3 MHz speed variant Rev 4.0 MPC106 device with a local PCI bus frequency of 33 MHz.

The main memory module is preferably implemented using JEDEC-standard SDRAM which permits single-cycle burst beats at 83.3 MHz bus frequency and is soldered directly to the SBC. An 8-bit wide device configuration will be employed to minimise address and control signal loading. If necessary, dependent on the precise implementation, bi-directional data registers may be provided between the processor/PCI bridge and the main memory to ensure reliable operation at the preferred frequency of 83.3 MHz.

Technical specifications and user manuals are readily available for the MPC750 processor and MPC106 PCI bridge/memory controller.

The level 2 cache 13 is preferably a pipeline burst SRAM device soldered direct to the SBC.

The Intel 21554 PCI-PCI bridge 5 is a non-transparent bridge with 64-bit primary and secondary PCI interfaces. The 64-bit secondary interface can only operate at 32-bit transfers in either direction since all other components on the local PCI bus 6 are 32-bit devices. The 64-bit primary interface may, however, provide, performance benefits in applications in which the PCI back-plane 7 is a 64-bit back-plane, otherwise it will simply operate as a 32-bit device.

In summary, the present invention provides a system architecture which allows very low cost CPU boards to be constructed exclusively through the use of standard components, the processing power of which can be readily scaled by adding additional boards (taking advantage of PCI architecture scalability) whilst retaining 100% memory access between CPUs.

CLAIMS

1. A multi-processor computer system comprising one or more computer modules connected to a host computer via a common PCI bus system backplane, the host computer comprising a host microprocessor and associated memory unit, and the or each computer module comprising:

a PCI connector connected to the PCI backplane;

at least one CPU;

a PCI backplane interface installed between the PCI connector and the or each CPU for forwarding PCI memory cycles to and from the PCI backplane; and

memory address translating means for translating memory cycles forwarded from the PCI backplane;

wherein the or each CPU comprises;

a local microprocessor having an associated local memory unit installed on a local memory bus;

a bridge installed between the local memory bus and the PCI interface of the respective computer module;

wherein said CPU bridge includes an address decoder which is programmed to a local address range for access to the respective computer module address space including the local memory unit associated with the or each CPU of that module;

wherein said PCI interface is programmed to forward PCI memory cycles from the PCI backplane which fall within an address range which is different from the local address range of the or each CPU of that module and which is different for each computer module in the system;

and wherein said translating means operates to translate memory cycles falling within said address window into corresponding addresses of the local address space of the respective computer module such that addresses corresponding to a local memory unit of that module may be decoded by the respective CPU bridge decoder;

whereby each microprocessor of the system can access the entire PCI memory space comprising the memory space of each computer module including the local memory units of all microprocessors of the system.

2. A multi-processor computer system according to claim 1, wherein the address translation means of each computer module is provided by the respective PCI backplane interface.

3. A multi-processor computer system according to claim 2, wherein the address translating means comprises appropriately programmed base address registers of the PCI backplane interface.

4. A multi-processor computer system according to any preceding claim, wherein the PCI address window of the or each computer module is an address window corresponding in size to the available memory space, including local micro-processor memory units, of the respective module.

5. A multi-processor computer system according to any preceding claim, wherein the PCI backplane interface is a PCI bridge.

6. A multi-processor computer system according to claim 5, wherein the PCI interface is a non-transparent PCI bridge.

7. A multi-processor computer system according to claim 5 or claim 6, wherein the or each CPU of a respective module is installed on a local PCI bus and the respective PCI bridge is a PCI-PCI bridge installed between the PCI backplane and the local PCI bus.

8. A multi-processor computer system according to claim 7, wherein said local PCI bus of at least one or the or each computer modules supports one or more PCI sites accessed by PCI addresses which contribute to the local memory space of the respective computer module.

9. A multi-processor computer system according to claim 7 or claim 8, wherein the or each CPU bridge interfaces with the respective computer module local PCI bus.

10 A multi-processor computer system according to any preceding claim, wherein the or each microprocessor is a PowerPC processor.

11. A multi-processor computer system according to claim 10 when dependent from claim 9, wherein the or each CPU bridge is a Motorola MPC106 bridge or the like, and the or each computer module further includes at least one decoding device installed on the respective local PCI bus which is programmed to decode PCI memory cycles appearing on the local PCI bus and is operable on identification of a PCI address corresponding to a physical address of a local CPU memory unit of that computer module to assert signal "ISA Master" on the respective CPU bridge thereby causing said bridge to decode the memory cycle as a respective local memory unit access.

12. A multi-processor computer system according to claim 11, wherein the or each decoding device is a PCI-compliant programmable logic device.

13. A multi-processor computer system according to any preceding claim, wherein the PCI interface is provided by an Intel 21554 PCI-PCI bridge or the like.

14. A multi-processor computer system according to any preceding claim, wherein the local address range, or local address ranges, is the same for the or each computer module of the system.

15. A multi-processor computer system comprising one or more computer modules connected to a host computer via a common PCI bus system backplane, the host computer comprising a host microprocessor and associated memory unit, and the or a computer module comprising:

a PCI connector;

at least one CPU;

a PCI backplane interface installed between the PCI connector and the or each CPU operable for forwarding PCI memory cycles to and from the PCI connector;

memory address translating means operable for translating memory cycles forwarded from the PCI connector;

wherein the or each CPU comprises;

a local microprocessor having an associated local memory unit installed on a local memory bus;

a bridge installed between the local memory bus and the PCI interface;

wherein said CPU bridge includes an address decoder which is programmed in use to a local address range for access to the respective computer module address space including the local memory unit associated with the or each CPU;

wherein said PCI interface is programmed in case to forward PCI memory cycles from the PCI connector which fall within an address range which is different from the local address range of the or each CPU of that module;

and wherein said translating means operates to translate memory cycles falling within said address window into corresponding addresses of the local address space of the respective computer module such that addresses corresponding to a local memory unit of that module may be decoded by the respective CPU bridge decoder.

16. A multi-processor computer system substantially as hereinbefore described, with reference to the accompanying drawing.

The Patent Office

INVESTOR IN PEOPLE

Application No: GB 9916289.3
Claims searched: 1-16

Examiner: Ben Micklewright
Date of search: 15 March 2000

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.R): G4A (AFGDC AFGL AMP)

Int Cl (Ed.7): G06F (13/38 13/40 15/16)

Other: Online: WPI, EPODOC, PAJ, INSPEC, COMPUTER

Documents considered to be relevant:

Category	Identity of document and relevant passage		Relevant to claims
X	GB2329984 A	(THOMSON) See whole document, e.g. pages 4-8	1-15
A	WO97/19405 A1	(S MOS) See whole document, e.g. pages 2,3	-
A	US5848249	(GARBUS) See whole document, e.g. column 8	-

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.